REMARKS

The foregoing amendment is provided to impart precision to the claims and to place them in condition for allowance, by more particularly pointing out the invention, rather than to avoid prior art.

Applicant respectfully requests reconsideration of the above identified application. Claims 1-30 are pending. Claims 1-30 are rejected. Claim 21 is amended.

Applicant respectfully notes that in the Office Action mailed on March 9, 2004, interpretations or characterizations by the Examiner, include inferences and/or potential limitations, to which Applicant does not agree. Being respectful of the Examiner's time, Applicant will not address all of such interpretations or characterizations in the present remarks but reserves the right to present further argument if the Examiner should rely upon them in the future.

The remaining comments are directed to Claim 1-30.

35 U.S.C. § 112 REJECTIONS

The Office Action mailed on March 9, 2004 rejects Claims 1 and 5-6 under 35 U.S.C. 112, second paragraph, as allegedly being indefinite, stating that it is not clear how the term "initialize" in independent claim 1 is distinct from or broader than the term "compute" in dependent claim 5 (and claim 6). Applicant respectfully offers the following reply.

The present application discloses, for example, that, "For one possible embodiment, an assertion graph, G, can be defined on a finite nonempty set of vertices, V, to include an *initial vertex, vI*;..." (p. 9, par. 2, lines 11-12, emphasis added) and "When an outgoing edge, e, originates from a vertex, v, and terminates at vertex, v', the original vertex, v, is called the head of e (written v = Head(e)) and the terminal vertex, v', is called the tail of e (written v' = Tail(e))" (p. 9, par. 2, lines 16-18).

The present application further discloses, for example, that, "To say that the model M satisfies assertion graph G in n steps (denoted by M $\models^n$ G), means that for any edge e *beginning at initial vertex vI* in G, all states, s, in M satisfy edge e in n steps" (p. 12, par. 2, lines 7-9, emphasis added).

The present application also discloses that, "For one embodiment, a simulation relation sequence can be defined for model checking according to the strong satisfiability criteria defined above. For an assertion graph G and a model M=(Pre, Post), define a simulation relation sequence, Sim_n: E→P(S), mapping edges between vertices in G into state subsets in M as follows:

$Sim_1(e)$ = Ant(e) if Head(e)=vI, otherwise

$Sim_1(e)$ = { };" (p.16, line 22 through p. 17, line 2).

The present application goes on to explain by way of example, with regard to Figure 3a, that, "Box 311 represents *initially assigning an empty set to the simulation relation* for all edges e in the assertion graph that do not begin at initial vertex vI, *and initially assigning Ant(e) to the simulation relation* for all edges e that do begin at initial vertex vI" (p.17, par. 2, lines 14-17, emphasis added).

The present application also explains by way of example, with regard to symbolic lattice domains and Figure 12a, that, "Box 1211 represents *initially assigning*

$Z = (initE(\underline{v}, \underline{v}') \wedge U) \cap_S Ant_S(\underline{v}, \underline{v}')$

to the simulation relation for all edges $(\underline{v}, \underline{v}')$ in the assertion graph that do not begin at initial vertex vI, and *initially assigning*

$Ant_S(\underline{v}, \underline{v}') = (initE(\underline{v}, \underline{v}') \wedge U) \cap_S Ant_S(\underline{v}, \underline{v}')$

to the simulation relation for all edges $(\underline{v}, \underline{v}')$ that do begin at initial vertex vI" (p.31, lines 10-15, emphasis added).

Applicant respectfully submits that at least in light of the examples presented above from the present application disclosure, the phrase "to initialize a symbolic simulation relation for an assertion graph," should not be held indefinite.

The definiteness of the language employed must be analyzed - not in a vacuum, but always in light of the teachings of the prior art and of the particular application

disclosure as it would be interpreted by one possessing the ordinary level of skill in the pertinent art. In re Moore, 439 F.2d 1232, 1235, 169 USPQ 236, 238 (CCPA 1971).

With regard to claims 5 and 6, the present application further explains by way of example, with regard to Figure 3a, that, "Box 313 represents testing the assertion graph to identify any active edges... Box 315 represents *recomputing the simulation relation* for edge, e, by adding to the simulation relation for edge e, any states which are in both the antecedent set for edge e and the post-image set for the simulation relation of any incoming edge, e', to e. Box 316 represents testing the simulation relation for edge e to determine if it was changed by the recomputation. If it has changed, all outgoing edges from e are marked as active, as represented by Box 317. In any case, the method flow returns to the test for active edges represented by Box 313" (p.17, line 18 through p. 18, line 2, emphasis added).

The present application also explains by way of example, at least with regard to symbolic lattice domains and Figure 12a, that, "Box 1215 represents *recomputing the simulation relation* for edge ($\underline{v}$, $\underline{v}'$) by adding to the simulation relation for edges ($\underline{v}$, $\underline{v}'$), any states which are in both the antecedent set for edges ($\underline{v}$, $\underline{v}'$) and the post-image set for the simulation relation of any incoming edges ($\underline{v}\,\bar{}$, $\underline{v}$) to ($\underline{v}$, $\underline{v}'$) produced by substituting any $\underline{b}$ in B^m for $\underline{v}\,\bar{}$. Box 1216 represents testing the simulation relation labeling for edges ($\underline{v}$, $\underline{v}'$) to determine if it was changed by the recomputation. If it has changed, the method flow returns to the recomputation of simulation relation for edges ($\underline{v}$, $\underline{v}'$), represented by Box 1215. Otherwise a fixpoint has been reached and the method terminates at box 1216" (p.31, lines 15-24, emphasis added).

Applicant respectfully submits that at least in light of the examples presented above from the present application disclosure, the phrase "to compute the symbolic simulation relation for the assertion graph," should not be held indefinite.

Applicant intends that embodiments of the claimed invention are described above and in the present application disclosure by way of example and not limitation. At least in light of the above teachings of present application disclosure, Claims 1, 5 and 6 should not be held indefinite. Accordingly, Applicant requests the Examiner withdraw the rejection under 35 U.S.C. 112.

35 U.S.C. § 102(b) REJECTIONS

The Office Action rejects Claims 1-30 under 35 U.S.C. 102(b) as allegedly being anticipated by Jain.

With regard to **Claim 1**, for example, the Office Action states that Jain discloses that the specification is defined as a set of abstract assertions defining the effect of each operation on the user visible state and an implementation mapping is used to relate abstract states to detailed circuit states. Applicant respectfully disagrees and offers the following arguments.

Jain claims that in the work of D. L. Beatty [13][15], three properties (Obedience, Conformity and Distinction) were used to verify all possible execution sequences, where the Obedience property verifies that individual abstract assertions hold for the circuit under the implementation mapping (§1.4.1, par. 1; §7.1, par. 1). Jain admits that his thesis concentrates on just the Obedience property (§1.4.1, par. 4).

Jain states that his approach supports the claim that verifying each individual abstract assertion amounts to verifying the entire abstract specification, and that he has developed a notion of stitching assertions together to reason about execution sequences (§1.2, par. 2).

Both the abstract specification and the implementation mapping of Jain are provided by the user (§9.1, par. 3). A main machine is a component of Jain's implementation mapping that defines the flow of control for individual system operations (§3.3, par. 2-3). Individual instructions can be stitched together in the implementation mapping by composing main machines to create execution sequences (§3.3.2, par. 3, Fig. 3.9). Even Jain admits that the implementation mapping might seem to be too complex (§8.6, par. 2).

Claim 1, on the other hand, sets forth a computer software product... having executable instructions... which, ...causes the processing device to initialize a symbolic simulation relation for an assertion graph on a first symbolic lattice domain.

Using an assertion graph, properties may be conveniently specified at various levels of abstraction. Intuitively, a model of a circuit or other finite state system can be simulated and the behavior of the model can be verified against properties expressed in an

assertion graph language. Important characteristics of this verification system are the expressiveness of the assertion graph language and the computational efficiency of carrying out the verification. When a property or properties are expressed in the assertion graph a processing device may automatically initialize a simulation relation to carry out the verification, for example, as described above with regard to the Examiner rejection of Claim 1 under 35 U.S.C. 112.

Jain does not disclose an assertion graph and does not cause a processing device to automatically initialize the simulation relation. Rather, Jain verifies that the individual abstract assertions hold for the circuit under the implementation mapping and discusses how these individual verifications can be stitched together for verifying the Obedience property in the implementation mapping, which even Jain admits might seem to be too complex. Moreover, both abstract assertions and implementation mapping are provided by the user to initialize the simulation.

Applicant, therefore, submits that causing a processing device to initialize a symbolic simulation relation for an assertion graph as set forth in Claim 1 is not anticipated by the cited reference.

With regard to **Claim 9**, set forth is a method comprising initializing a symbolic simulation relation for an assertion graph on a first symbolic lattice domain.

As stated above, Jain does not disclose an assertion graph but verifies that the individual abstract assertions hold for a circuit under an implementation mapping.

Applicant, therefore, submits that initializing a symbolic simulation relation for an assertion graph as set forth in Claim 9 is not anticipated by the cited reference.

With regard to **Claim 16**, a method comprising specifying a justification property with an assertion graph is set forth.

As stated above, Jain does not disclose an assertion graph. Further, Jain admits that his thesis concentrates only on the Obedience property (§1.4.1, par. 4), which permits complex safety properties that can check for future possibilities based on past or present state conditions. This capability is referred to as implication.

It is desirable to ask why a set of state conditions occurred. In other words, what possible initial conditions and transitions could cause the system under analysis to end up in a given state? Such a capability is referred to as justification. Assertion graphs of the

present application are capable of expressing justification properties and accordingly, Claim 16 sets forth specifying a justification property with an assertion graph.

Applicant, therefore, submits that specifying a justification property with an assertion graph as set forth in Claim 16 is not anticipated by the cited reference.

With regard to **Claim 21** as amended, set forth is a formal verification method comprising: defining a property as an assertion graph…; initializing a simulation relation from the assertion graph …; simulating a finite state system… to generate a subsequent state condition or an output of the simulation relation….

As stated above, Jain does not disclose defining a property as an assertion graph and does not use an assertion graphs to initialize the simulation relation.

Applicant, therefore, submits that defining a property as an assertion graph and initializing a simulation relation from the assertion graph as set forth in Claim 21 is not anticipated by the cited reference.

With regard to **Claim 22**, set forth is a verification system comprising means for initializing a symbolic simulation relation for an assertion graph on a first symbolic lattice domain.

As stated above, Jain does not disclose means for using an assertion graph for initializing a symbolic simulation relation.

Applicant, therefore, submits that means for initializing a symbolic simulation relation for an assertion graph as set forth in Claim 22 is not anticipated by the cited reference.

With regard to **Claim 29**, set forth is a verification system comprising means for specifying a justification property with an assertion graph.

As stated above, Jain does not disclose an assertion graph. Further, Jain's thesis concentrates only on the Obedience property (§1.4.1, par. 4), which can only check for future possibilities based on past or present state conditions.

Assertion graphs of the present application are capable of expressing justification properties.

Applicant, therefore, submits that means for specifying a justification property with an assertion graph as set forth in Claim 29 is not anticipated by the cited reference.

Therefore, Applicant respectfully submits that Claims 1, 9, 16, 21, 22 and 29 are patently distinguished over the art cited by the Examiner. Applicant further believes that Claims 2-8, 10-15, 17-20, 23-28 and 30 being dependent therefrom are also patentable. Applicant respectfully requests the Examiner withdraw his rejection under 35 U.S.C. 102(b).

Applicant, therefore, believes that Claims 1-30 are presently in condition for allowance and such action is earnestly solicited.
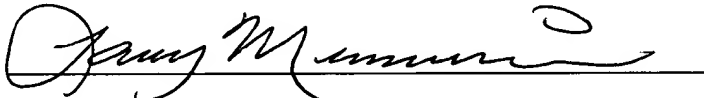
CONCLUSION

Applicant respectfully submits the present claims for allowance. If the Examiner believes a telephone conference would expedite or assist in the allowance of the present application, the Examiner is invited to call Lawrence M. Mennemeier at (408) 765-2194.

Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN

Date: 9-9-2004

Lawrence M. Mennemeier

Reg. No. 51,003

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(408) 720-8300